Installation Guide

General Setup - Web Server

This project expects the user to have access to a running web server. In our case specifically, we tested using the UML CS mercury server as well as locally using XAMPP on a Windows machine. There are many other resources available on setting up web servers and how to do so is beyond the scope of this project.

Install Example - cs.uml.edu servers (Linux Install)

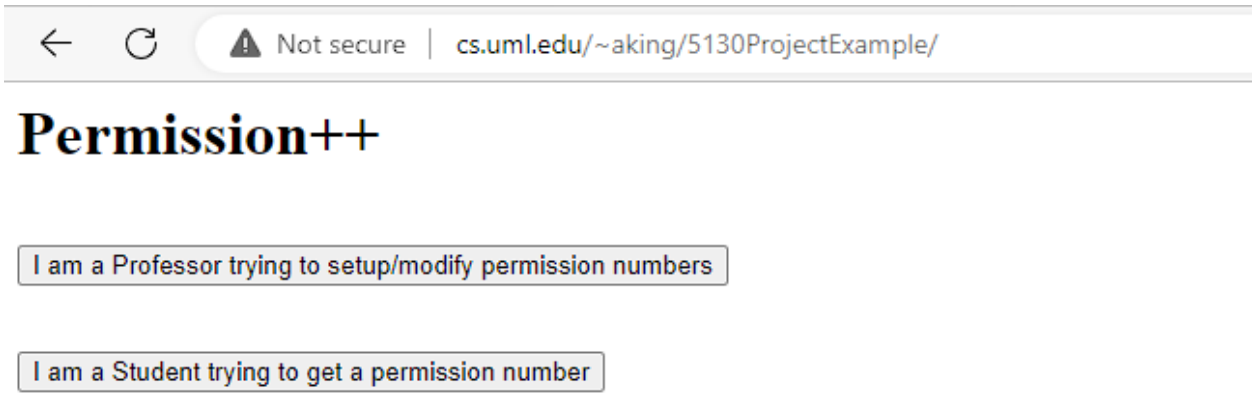
First go to https://github.com/adk3798/5130f2022/tree/main/King-Adam-Project. Copy all files in that directory into a directory within the directory that is outside accessible on your web server. In my case, this meant creating a directory inside the “public_html” directory in my home directory of my UML CS account (a screenshot showing the file setup is included later). Next, you must set the permissions. The proper way to set the permissions would be to change the ownership of the directory you created with all the project files to be the web server and setting the permissions so that the owner can read/write/execute the files but other users can not (700 permissions with correct ownership basically). However, I do not have sudo privileges on the cs.uml.edu servers so I am not allowed to chown files (even those I own). For testing purposes here, it is simpler to just set the permissions on the directory to just be 777 which will guarantee the web server will be able to do all it needs (and others will technically be able to as well, but this is just for testing). Here is an example of the file structure, including the permission, within my pubic_html directory that worked.

```
aking@cs1:~/public_html$ ls -l
total 8
drwxrwxrwx 3 aking cs2020 4096 Dec  2 17:54 5130ProjectExample
drwxrwxrwx 3 aking cs2020 4096 Dec  2 17:46 App
aking@cs1:~/public_html$ ls -l 5130ProjectExample/
total 96
-rw-r--r-- 1 aking cs2020   972 Dec  1 14:01 addClass.php
-rw-r--r-- 1 aking cs2020  2257 Dec  1 14:16 addNumbers.php
-rw-r--r-- 1 aking cs2020   475 Dec  1 20:03 checkAccountExists.php
-rw-r--r-- 1 aking cs2020  1398 Dec  1 16:25 checkByEmail.php
-rw-r--r-- 1 aking cs2020  1384 Dec  1 16:25 checkByID.php
-rw-r--r-- 1 aking cs2020   486 Dec  1 19:50 checkLogin.php
-rw-r--r-- 1 aking cs2020   925 Dec  1 14:59 clearNumbers.php
-rw-r--r-- 1 aking cs2020  2802 Dec  1 20:14 createAccount.html
-rw-r--r-- 1 aking cs2020   421 Dec  1 19:22 createAccount.php
drwxr-xr-x 2    48     48  4096 Dec  2 17:55 data
-rw-r--r-- 1 aking cs2020  2503 Dec  1 17:53 freeNumbers.php
-rw-r--r-- 1 aking cs2020   865 Dec  1 20:17 index.html
-rw-r--r-- 1 aking cs2020   381 Nov 30 19:46 listClasses.php
-rw-r--r-- 1 aking cs2020   373 Nov 30 07:15 profPage.css
-rw-r--r-- 1 aking cs2020 12507 Dec  1 19:56 profPage.html
-rw-r--r-- 1 aking cs2020 44133 Oct 13 17:32 Project_UI.png
-rw-r--r-- 1 aking cs2020   105 Oct 13 17:36 README.txt
-rw-r--r-- 1 aking cs2020   732 Dec  1 14:30 rmClass.php
-rw-r--r-- 1 aking cs2020  2053 Dec  1 14:17 rmNumbers.php
-rw-r--r-- 1 aking cs2020   214 Nov 22 09:33 student.css
-rw-r--r-- 1 aking cs2020 10479 Dec  1 16:32 student.html
-rw-r--r-- 1 aking cs2020  2557 Dec  1 16:18 student.php
-rw-r--r-- 1 aking cs2020  1426 Dec  1 15:36 viewNumbers.php
-rw-r--r-- 1 aking cs2020    10 Oct 13 17:37 _whoAmI.txt
```

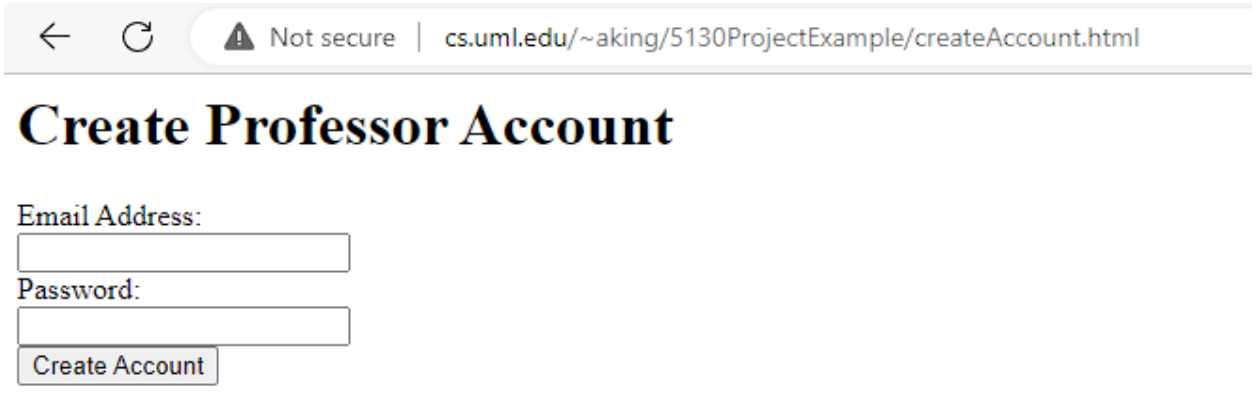
There are a few here you can ignore for now. First is the “App” directory in the first ls command output. That is unrelated to the project. Second is the “data” directory inside the 5130ProjectExample directory. That directory will likely not exist yet (it will be created for you later, I didn’t take this screenshot until I was one step further in). Lastly, there may be some pdf files I later include (including this install guide I am writing here) that may have been copied over. These files have no effect on functionality and can be ignored. Otherwise, the setup should be the same. Note that all the files from the linked Github directory are present in the directory I made for the project and that the permissions on the directory containing the files is 777 (done with just a “chmod 777 5130ProjectExample” command). Again, I don’t have the necessary permissions to properly set this up, which, in this case, would be setting the owner of the directory to 48 (the Apache web server) rather than changing the permissions. This will work fine for testing, but on a real system (where I expect you’d have permissions) that’s what you’d do.

Once the files are installed and the permissions set, navigate to the project homepage, which should just be cs.uml.edu/~<your-username>/<project-directory>. For example, for me it is

“cs.uml.edu/~aking/5130ProjectExample”. Navigating to this page should get you something like what is shown below.

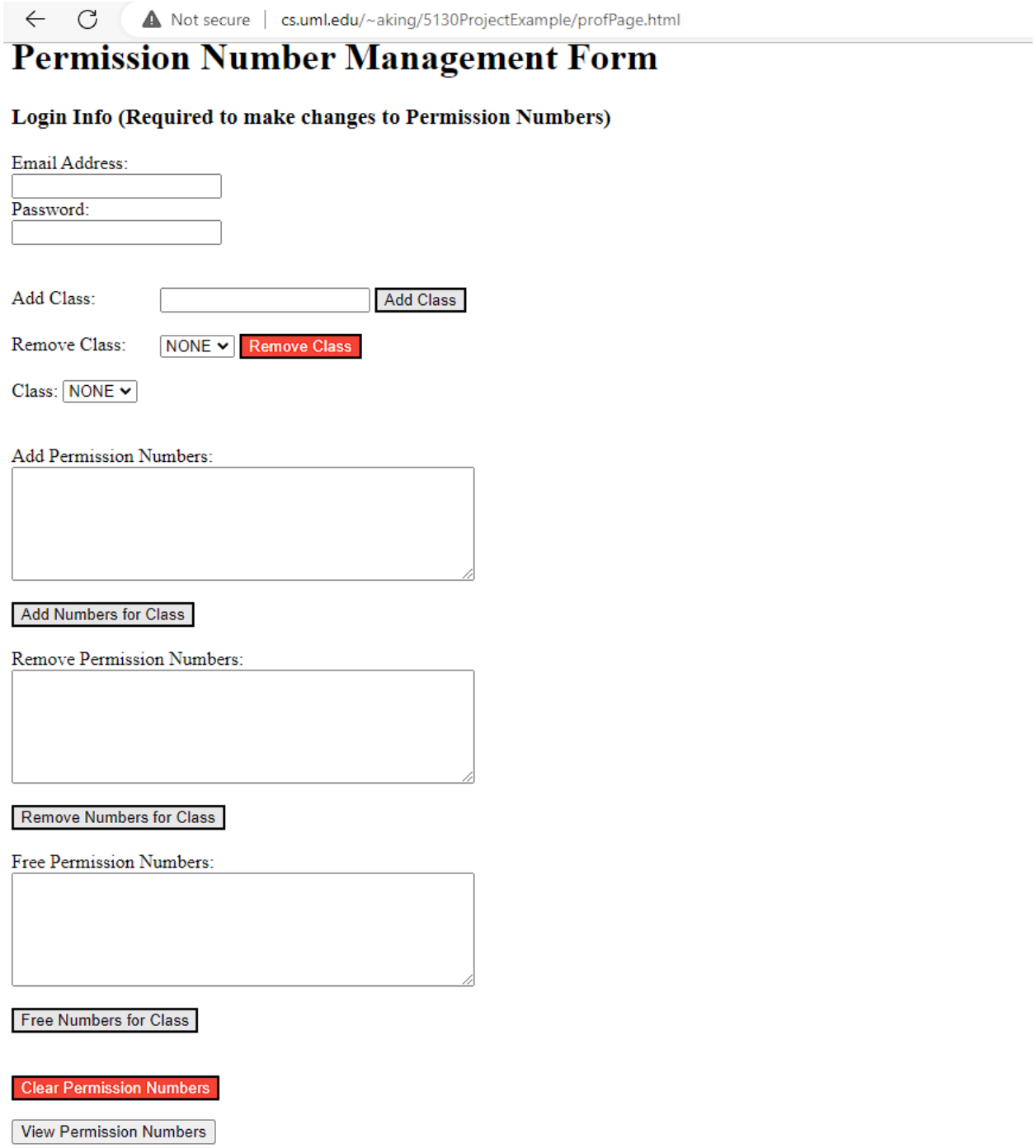


What you will want to do from there is click the first button specifying you are a professor. This should bring you to a screen like the screenshot below. This page will only be available when no account exists yet. The idea is to just create an account to use for the class sign up period and tear it down later, so I’d recommend storing the login info somewhere. One limitation of the project is it currently doesn’t have functionality for password changes or account recovery so make sure it’s something you’ll remember.



I personally created an account with email “me@me.com” and password “password” (obviously use your real email and a more secure password in practice). Once you’ve entered in your information (it requires a technically valid email and at least 8 chars for the password) hit the

“Create Account” button. This should bring you to the “professor page” if it works properly (if it doesn’t it’s most likely the file permissions causing the problem).



As a last check that account creation succeeded, check the directory you created the project files in again. There should now be a “data” directory in there as there was in the first screenshot. Additionally, if you check inside that data directory, you should see a login.json file. Catting out this file will give your login email as well as a hash of your password. It’s not part of

normal operation, but in case things get stuck, removing that file will allow you to create another account. Here is what it looked like for me.

```
aking@cs1:~/public_html$ ls -l App/data/
total 1
-rw-r--r-- 1 48 48 112 Dec  2 17:40 login.json
aking@cs1:~/public_html$ cat App/data/login.json
{
    "email": "me@me.com",
    "password": "5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8"
}aking@cs1:~/public_html$
```

That file's existence should confirm the installation has worked properly and from here on all actions should be possible through the web UI rather than manually. Do note that the ownership of files in the data directory is user "48" which is the user for the Apache web server on the cs.uml.edu servers. This means these files will not be writable or executable by other users (unless they have sudo access).

All further operations will be included in the **user manual**, which should only require interaction through a web browser. It will be assumed in the user manual that you know the login information for the account created during the steps in this install guide..